NIST Open Machine Translation 2009 Evaluation: Tel Aviv University's System Description

Kfir Bar and Nachum Dershowitz
Computer Science Department
Tel Aviv University

1 Site Affiliation

Tel Aviv University

2 Contact Information

Kfir Bar
Dept. of Computer Science
Tel Aviv University
Israel
kfirbar@post.tau.ac.il

3 Submission

TLVEBMT_a2e_cn_primary

4 Primary System Specs

4.1 Single System Vs. System Combination

Single System

4.2 Core MT Engine Algorithmic Approach

This document describes our first participation in the NIST MT evaluation project. We submitted the results of a non-structural Example-Based Machine Translation system that translates text from Arabic to English, using a parallel corpus aligned at the paragraph / sentence level. Each new input sentence is fragmented into phrases and those phrases are matched to example patterns, using various levels of morphological information.

The system exploits a bilingual corpus to find examples that match fragments of the input source-language (Modern Standard Arabic-MSA, in this case) text, and imitates its translations. In the matching step, the system uses various levels of morphological information in order to broaden the amount of matched translation examples and to generate new translations based on morphologically similar fragments. In addition, we forced the matching algorithm to work on the phrase level only. The operant definition of a phrase for us is a combination of adjacent base-phrases of the input sentence.

In the transfer step, those matched phrases are translated using the target-language (English, in our case) version of the parallel corpus, using the automatically created translation example alignment table.

In the recombination step all the translated fragments are pasted together to form a complete target-language text, usually by preferring larger translated fragments since they use more context.

4.2.1 Parallel Data

The translation examples in our system were extracted from a collection of parallel unvocalized Arabic-English corpora, all provided by the LDC. In the current submission, we used the Arabic English Parallel News Text Part 1 (LDC2004T18), the Arabic News Translation Text Part 1 (LDC2004T17), the Arabic Treebank English Translation (LDC2005E46), the Arabic Treebank: Part 1 - 10K-word English Translation (LDC2003T07) and test data from previous NIST MT evaluation (2004 - LDC2006E44, 2005 - LDC2006E39).

All the translation examples were morphologically analyzed using the Buckwalter morphological analyzer (version 1.0) (Buckwalter, 2002), and then part-of-speech tagged using AMIRA (Diab et al., 2004). For each Arabic word in the translation example, we look up its English equivalents in a lexicon, created using the Buckwalter glossaries, and then expand those English words with synonyms from WordNet. Then we search the English

version of the translation example for all instances on the lemma level and insert them in a special alignment table.

The Arabic version of the corpus was indexed on the word, stem and lemma levels (stem and lemma, as defined by the Buckwalter analyzer), so, for each given word, we are able to retrieve all translation examples that contain that word on any of those three levels.

4.2.2 Matching

Given a new input sentence, the system begins by searching the corpus for translation examples for which the Arabic version matches fragments of the input sentence. In the implementation we are describing, the system is restricted to fragmenting the input sentence so that a matched fragment must be a combination of one or more complete adjacent base-phrases of the input sentence. The base-phrases are initially extracted using the AMIRA tool. Fragments also must contain at least two words. For instance, take the following sentence:

يكون المعهد قادرا على القيام ببحوث مستقلة

(*Ykwn AlmEhd qAdrA ElY AlqyAm bbHwv mstqlp*, "The institute is able to pursue independent research"). Its AMIRA base-phrases are:

[VP *ykwn*] [NP *AlmEhd*] [ADJP *qAdrA*]
[PP *ElY AlqyAm*] [PP *bbHwv mstqlp*]

That means, for example, that the fragment *ykwn AlmEhd qAdrA* is possible, but the fragment *ElY AlqyAm bbHwv* is not allowed, because it is not a combination of complete adjacent base-phrases. Note that matching the complete input sentence is allowed. Currently, we have not taken the types of base-phrases into consideration, but it seems that using this kind of information, compiled into several pattern rules (e.g. matching the sequence PP NP), will improve the matching results, by forcing the system to only consider reasonable sequences of base-phrases.

The same fragment can be found in more than one translation example. Therefore, a "match-score" is assigned to each fragment-translation pair, signifying the quality of the matched fragment in the specific translation example.

Fragments are matched word by word, so the score for a fragment is the average of the individual word match-scores. To deal with data sparseness we generalize the relatively small corpus by matching words on text, stem, lemma, and morphological levels, with each level assigned a different score. Text (exact string) and stem matches credit the words with the maximum possible. The lemma of a word is revealed using the Buckwalter analyzer, and matching words on that level credits them with fewer points. The morphological level credits the fragment match-score with a minimal amount. Text and stem matches receive almost the same score, since, currently, we do not yet handle the translation modification needed. When dealing with unvocalized text, there are, of course, complicated situations when both words have the same unvocalized stem but different lemmas, for example, the words كتب (*katab*, "wrote") and كتب (*kutub*, "books"). Such cases are not yet handled accurately, since we have not worked with a context-sensitive Arabic lemmatizer and so cannot derive the correct lemma of an Arabic word. Still, the combination of the Buckwalter morphological analyzer and the AMIRA part-of-speech tagger allows us to reduce the number of possible lemmas for every Arabic word so as to reduce the amount of ambiguity. Actually, by "lemma match", we mean that words match on any one of their possible lemmas. Further investigation, as well as working with a context-sensitive morphology analyzer (Habash and Rambow, 2005), will allow us to better handle all such situations. In the current implementation we used additional two levels: cardinal, for matching numeric tokens, and proper-noun, for matching words that represent named entities. Named entities in our implementation are those words that were tagged with the *NNP* tag by AMIRA part-of-speech tagger.

4.2.3 Transfer

The input to the transfer step consists of all the collected fragments found in the matching step, and the output is the translations of those fragments. The translation of a fragment is taken to be the best generated translation among the comprised fragments. Translating a fragment is done in two main steps: (1) extracting the translation of the example pattern from the English version of the translation example; (2) fixing the extracted translation so that it will be the translation of the fragment's source pattern.

First Step – Translation Extraction

The first step is to extract the translation of the fragment's example pattern from the English version of the translation example. Here we use the

prepared alignment table for every translation example within our corpus. For every Arabic word in the pattern, we look up its English equivalents in the table and mark them in the English version of the translation example. Then, we extract the shortest English segment that contains the maximum number of equivalent words. Usually a word in some Arabic example pattern has several English equivalents, which makes the translation extraction process complicated and error prone. For this reason, we also restrict the ratio between the number of Arabic words in the example pattern and the number of English words in the extracted translation, bounding them by a function of the ratio between the total number of words in the Arabic and English versions of the translation example.

For example, take the following translation example:

A: الخدمات الاستشارية والتعاون التقني في ميدان حقوق الإنسان

E: "Advisory services and technical cooperation in the field of human rights."

Table 1 is the corresponding alignment table. Now, suppose the example pattern is ميدان حقوق الإنسان (*mydAn Hqwq Al<nsAn*, "the field of human rights"), so we want to extract its translation from the English version of the translation example. Using the extracted look-up, we mark the English equivalences of the pattern words in the translation example: "Advisory services and technical cooperation in the field of human rights", and then we extract the shortest English segment that contains the maximum number of equivalent words, viz. "field of human rights".

English	Arabic
Services	الخدمات
Advisory	الاستشارية
Cooperation	والتعاون
Technical	التقني
In	في
Field	ميدان
Rights	حقوق
Human	الإنسان

Table 1. Alignment table

This is, of course, a simple instance. More complicated ones would have more than one equivalent per Arabic word.

Second Step – Fixing the Translation

Recall that the match of a corpus fragment to the input fragment can be inexact: words may be matched on several levels. Exactly matched words are assumed to have the same translation, but stem or lemma matched words may require modifications (mostly inflection and prepositions issues) to the extracted translation. These issues are left for future work. Words matched on the part-of-speech level require a complete change of meaning. For example, take the input fragment مجلس الامن (*mjls AlAmn*, "the Security Council"), matched to the fragment مسؤولية الامن (*ms&wlyp AlAmn*, "the security responsibility") in some translation example. The words مجلس (*mjls*, "council") and مسؤولية (*ms&wlya*, "responsibility") are matched on their morphological features level (both are nouns). Assume that the extracted translation from the translation example is "the security responsibility", which is actually a translation of مسؤولية الامن (*ms&wlyp AlAmn*, "the security responsibility") and is not the translation of the input pattern at all. But, by replacing the word "responsibility" from the translation example with the translation of مجلس (*mjls*, "council") from the lexicon, we get the correct phrase: "the security council". The lexicon is implemented using the glossaries extracted from the Buckwalter morphological analyzer and expanded with WordNet synonyms as was explained above.

Sometimes the extracted translation contains some extra unnecessary words in the middle. Those words appear mostly because of the different structure of a noun-phrase in both languages. For example, consider the example, موضوع الامن الاقليمي (*mwDwE AlAmn AlAqlymy*), and its translation: "the subject of regional security". By extracting the translation of the pattern موضوع الامن (*mwDwE AlAmn*), we obtain: "the subject of regional security" (since it is the shortest segment that contains maximum word alignments). Clearly, the word "regional" is unnecessary in the translation because it is the translation of the word الاقليمي (*AlAqlymy*, "the regional") that does not appear in the pattern. So by removing that word from the translation we obtain the correct translation of the pattern. The word "regional" appears in the extracted translation due to the fact that Arabic adjectives come after the nouns they qualify, which is the opposite of English syntax. Here, the noun-

phrase الامن الاقليمي (*AlAmn AlAqlymy*, "the regional security") is translated so that the translation of الاقليمي (*AlAqlymy*, "the regional") appears before the translation of الامن (*AlAmn*, "security"). Currently, identifying such situations is done by searching for the translation of the word "regional" in a fixed number of Arabic words that come immediately after the pattern in the translation example.

Removing unnecessary words from the extracted translation must preserve the correct English syntax of the remaining translation, which in some cases seems to be a difficult task. This task is not in the scope of the submitted system and it is currently being investigated.

A translation-score is given for each translation fragment. The translation-score is calculated based on the ratio between the number of covered words and the total number of words in the Arabic pattern. The total-score of fragment is the average of its match-score and its translation-score.

4.2.4 Recombination

In the recombination step, we paste together the extracted translations to form a complete translation of the input sentence. This is generally composed of two subtasks. The first is finding the *N* best (where *N* is a configurable parameter) recombinations of the extracted translations that cover the entire input sentence, and the second is smoothing out the recombined translations to make a fully grammatical English sentence. Currently, we handle only the first subtask in which we output the recombinations obtaining the best cover over the given input source-language sentence. The best cover is obtained by preferring long translated fragments over covered short ones, as well as preferring covers composed of less fragments. Finding the *N* best covers is performed in a dynamic programming fashion. By multiplying the total-scores of the comprised general fragments, we calculate a final-translation-score for each generated recombination. The final translation is chosen among the set of the *N* best recombinations using the final-translation-score.

The final translation is post-processed for fixing several elementary English issues.

4.3 Critical Additional Features and Tools Used

We used several linguistic tools within our current implementation. WordNet is used in the process of building the alignment table for each translation example. We used it as an English stemmer as well as for retrieving all the relevant synonyms of an English word.

AMIRA is used as an Arabic part-of-speech and base-phrase tagger.

The Buckwalter morphological analyzer (version 1.0) was applied on the corpora's Arabic part and the input sentence as well.

We used BaseNP (Ramshaw and Marcus, 1995) chunker and Brill's part-of-speech tagger (Brill, 1992) on the final translation, as described in the next section.

4.4 Significant Data Pre/Post Processing

As mentioned, the final translation was post processed for fixing some critical issues. First, all doubled words / punctuation marks were eliminated. Then we converted the text to lower case. Words that were originally (in the translation example) written with a first capital letter were searched in Buckwalter's glossary list. In case it was not found, we changed it to lower case.

Final translations starting with a verb and a following noun-phrase were fixed by switching the places of those components. We used Brill's part-of-speech tagger and BaseNP chunker for this purpose.

4.5 Other Data Used

We did not use any external data outside the predefined NIST MT evaluation LDC list.

References

Brill, Eric. 1992. A Simple Rule-Based Part-Of-Speech Tagger. *In Proceedings of the DARPA, Speech and Natural Language Workshop*. 112-116. Morgan Kauffman. San Mateo, CA.

Buckwalter, Tim. 2002. Buckwalter Arabic Morphological Analyzer Version 1.0. *Linguistic Data Consortium*, Philadelphia, PA.

Diab, Mona, Kadri Hacioglu and Daniel Jurafsky. 2004. Automatic Tagging of Arabic Text: From Raw Text to Base Phrase Chunks. *The National Science Foundation*, Washington, DC.

Habash, Nizar and Owen Rambow. 2005. Arabic Tokenization, Morphological Analysis, and Part-of-Speech Tagging in One Fell Swoop. In: *Proceedings of the Conference of American Association for Computational Linguistics.*

Lance A. Ramshaw and Mitchell P. Marcus. 1995. Text Chunking Using Transformation Based Learning. In: *Proceedings of the Third ACL Workshop on Very Large Corpora.*